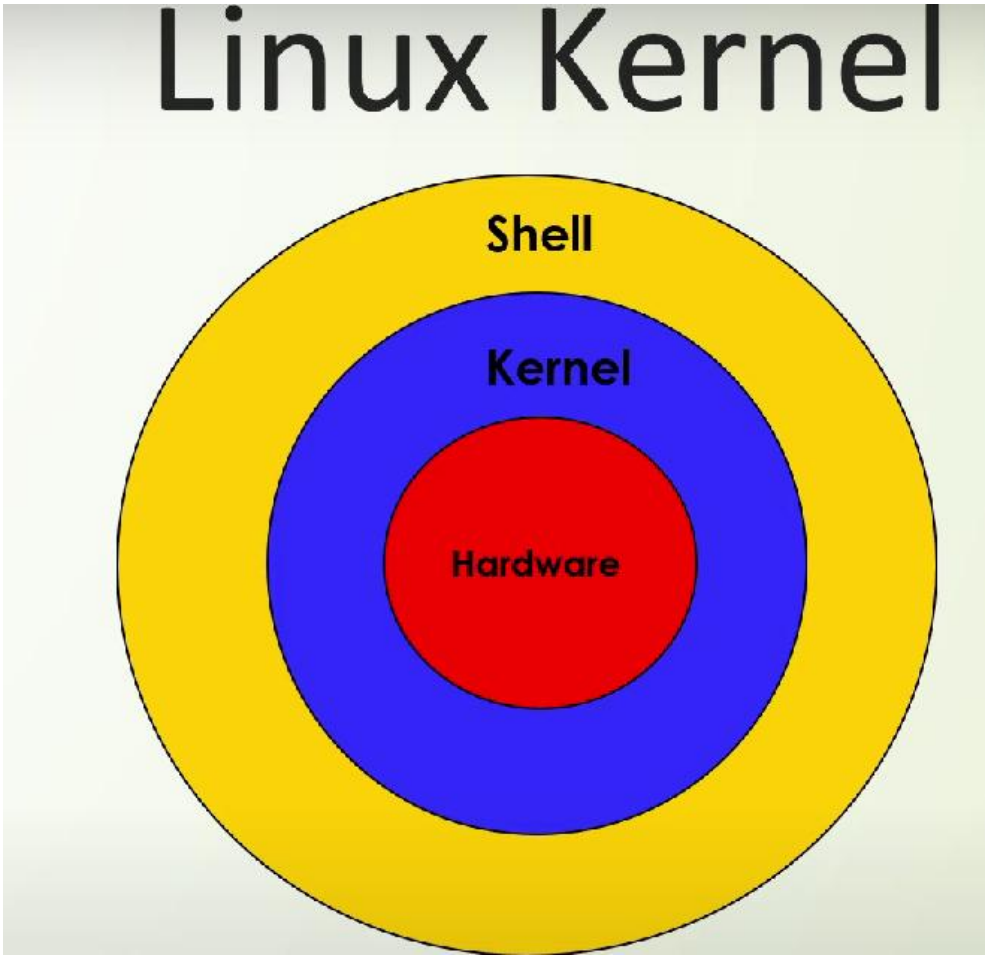
Kernel, Shell & Linux File System - Explained for Beginners

In-depth with Analogies and Examples

What is Linux kernel?



Linux Kernel

- It’s the heart of Linux. It connects the user and the hardware with the help of shell. Shell provides a interface where user can execute commands and with the help of these commands, the OS performs the tasks on the hardware with the help of kernel.
  Which means the kernel pass the instruction from the user to the hardware.

Shell ???

- Shell is the env where we excutes the commands on the terminal.
  We can compare shell as a command prompt in windows.
  In summary, we can say that the shell is powerful interface to the Unix/Linux OS, where you can manipulate data and execute several applications under certain conditions.
  -Also known as CLI.
  -Types- bash, zsh, sh etc.

How can you access the shell??

- To use the shell, you need to authenticate first (need to tell system who are you). This process is commonly called login process and in involves 2 steps.
  - You must have credentials to login. This will be provided by the root/admin.
- To check the shell: echo $SHELL

/ - Root - House Entrance

- This is the very beginning of the Linux file system. All files and directories stem from here, just like how your house begins at the entrance. It doesn't belong to any parent directory.
- Example: Everything is under /, like /home, /etc, /usr

/home - User Space - Your Bedroom

- Each user gets a personal folder in /home. Just like each family member has their own room. You store your personal documents, downloads, and settings here.
- Example: E.g., /home/jane/Documents or /home/raj/Desktop

/etc - System Settings - Control Panel

- Holds configuration files for your system and installed software. It's like adjusting your router, desktop settings, or app preferences.
- Example: E.g., /etc/hostname, /etc/fstab, /etc/apache2/

/bin - Essential Tools - Toolbox

- Contains core commands used by the system and users. These tools are available even in emergency situations (single-user mode).
- Example: E.g., /bin/ls (list files), /bin/cp (copy files)
- Use case: Commands like ls, cp, mv, rm.

/usr - Apps & Libraries - Library and App Store

- Contains user-installed software and libraries. Think of it as a shared app store or library used by all users.
- Example: E.g., /usr/bin/python3, /usr/lib, /usr/share/fonts

/tmp - Temporary Files - Scratch Paper

- Used by applications to store data temporarily. Files here are deleted when the system reboots. Like a whiteboard for quick work.
- Example: E.g., /tmp/install-log.txt

/var - Variable Data - Whiteboard / Logs

- Stores files that constantly change, such as logs, print jobs, and mail. It's like a shared whiteboard where the system updates info in real-time.
- Example: E.g., /var/log/syslog, /var/mail

/boot - Boot Loader - Power Switch

- Contains files needed to boot your Linux OS, including the Linux kernel and GRUB loader. Without this, your system won't start.
- Example: E.g., /boot/vmlinuz, /boot/grub/

/dev - Devices - Power Sockets

- Represents devices (like USBs, hard drives) as files. You can interact with hardware directly via these files.
- Example: E.g., /dev/sda (first hard disk), /dev/tty (terminals)

/proc & /sys - System Info - X-ray Scanner

- /proc and /sys are virtual filesystems that show system and kernel data. It’s like viewing your system’s internals in real time.
- Example: E.g., /proc/cpuinfo, /sys/class/net/eth0

/root - Admin’s Home - Boss’s Room

- This is the home directory of the root user. Like the owner's private office — only accessible to administrators.
- Example: E.g., /root/.bashrc